

A look-ahead trajectory planning algorithm for spray painting robots with non-spherical wrists

Giulio Trigatti¹, Paolo Boscariol², Lorenzo Scalera³, Daniele Pillan¹, and
Alessandro Gasparetto³

Abstract This paper proposes an innovative trajectory planning algorithm aimed at limiting the speed and accelerations for robots for spray painting. The algorithm can be applied to robots that, due to their kinematic structure, do not offer a closed-form solution to the inverse kinematic problem. The trajectory planning problem is cast for a predefined path, which can be described by a B-splines. Such path is then described by a set of discrete samples, that are filtered to achieve end-effector speed and joint acceleration constraints. The proposed method has been applied to a commercially available painting robot equipped with a non-spherical wrist. The experimental results show a reduction of the joint speed, acceleration, and a sensible reduction of the motor efforts.

Key words: trajectory planning, painting robot, B-splines, path planning

1 Introduction

In many robotic tasks, trajectories are usually programmed in the end-effector space, where the task is easier to define. This approach of motion planning has to take into account the transformation between the operative and the joint spaces, which in general is not linear. When it is necessary to limit accelerations in order to improve the quality of the robot tasks and to reduce the vibrations of the robotic arm, it is necessary to adopt a method that takes into account this nonlinearity. In fact, due to the nonlinearity of the robot kinematics, an effective reduction of accelerations in the operational space does not necessarily imply a reduction of the accelerations in joints space as well. For these reasons, in this paper we propose a methodol-

¹ CMA Robotics, Pavia di Udine (UD), Italy

² Università degli Studi di Padova, DTG, Vicenza, Italy
e-mail: paolo.boscariol@unipd.it

³ Università degli Studi di Udine, DPIA, 33100 Udine, Italy

ogy able to consider the relations between end-effector and joint spaces in order to limit the acceleration of joints directly in the joint space but capable of creating a suitable velocity profile in the operational space. This technique allows to obtain good results in terms of accelerations constraints and to achieve a simple description of the trajectory for control purposes. In the recent years, a large number of path planning methods have been developed in the field of speed and acceleration control for enhancing the performance of high-speed CNC machines in industrial applications. In particular, several look-ahead speed control algorithms based on linear acceleration profile have been proposed. Examples can be found in [1], where a look-ahead algorithm that considers speed limitation of each axis is proposed, in [2] and [3], where a velocity profile generation method and an optimal feedrate model for high-speed machining are described. Furthermore, the look-ahead strategy has been applied for the interpolation of continuous micro-line trajectories in [4] and [5]. To avoid the limitation of the linear acceleration profile, several algorithms based on Bezier [6] and Non-Uniform Rational B-Spline (NURBS) curves [7, 8] have been proposed. Another example is given in [9], where a method to define an optimal feedrate method of NURBS interpolator for CNC machines tools is described. However, the methodologies mentioned above, which give good results on machines with simple kinematics (such as CNC machines), are not suitable for robots, in particular for those with non-closed solution of inverse kinematics due to the presence of a non-spherical wrist. In order to address this problem, it is necessary to develop a more sophisticated method, based on a re-parametrization of the trajectory. A particularly complex case of machines with nonlinear kinematics is given by industrial painting robots [10, 11, 12]: most of those have a non-spherical wrist in order to avoid wrist singularities during painting tasks. This leads to a complex formulation of the inverse kinematics, which has no closed-form solution. In this paper, a trajectory planning technique, based on a look-ahead strategy, aimed at limiting the accelerations of joints in robots with non-spherical wrist, is presented. Unlike CNC machines, where the end-effector acceleration is linearly related to the acceleration of joints, the algorithm proposed here allows to create a velocity profile in the Cartesian space that can limit the accelerations in the joints space, as well as the motor torques at the joints.

2 The look-ahead algorithm

By assuming a geometric path described by the curvilinear abscissa u , the method described in the following defines a sequence of points in the end-effector space, such as the joint accelerations result below the imposed limits.

Given a parametric path in the end-effector space $s = \mathbf{B}(u)$, a sequence of points along the path can be obtained by defining:

$$u_{k+1} = u_k + \frac{v_t}{\left| \frac{d\mathbf{B}(u)}{du} \right|_{u_k}} T \quad (1)$$

where T is a suitable constant time interval between two consecutive points and v_t is the desired velocity. Once the sequence of the u_k has been obtained, the end-effector pose corresponding to any u_k can be computed. Then, the geometric path $\mathbf{B}(u)$ is executed at constant speed v_c and, through the inverse kinematics map, a sequence of joint positions corresponding to the u_k can be obtained. Hence, the resulting trajectory in the end-effector space is described by equal intervals of space $\Delta s = v_c T$. Knowing the joint position at any time step, the joint acceleration $\ddot{q}(k)$ at the k -th time step can be computed:

$$\ddot{q}_k = \frac{q_{k+1} - 2q_k + q_{k-1}}{T^2} \quad (2)$$

where q_k , q_{k+1} and q_{k-1} are the joint position coordinates at time steps $k+1$, k and $k-1$, respectively. Knowing the maximum admissible absolute value of the acceleration for each joint \ddot{q}_{lim}^i , we can determine the time interval \hat{T}^i which represents the minimum time to go from q_{k-1}^i to q_{k+1}^i at constant speed for the i -th joint. It results:

$$\hat{T}^i = \sqrt{\frac{q_{k+1}^i - 2q_k^i + q_{k-1}^i}{\ddot{q}_{lim}^i}} \quad (3)$$

The maximum value of the \hat{T}^i 's is used to calculate the tangential velocity $v_{t,k}$ at k -th time instant as:

$$v_{t,k} = v_c \frac{T}{\max_i \hat{T}^i} \quad (4)$$

However, the velocity profile thus obtained cannot be used as it is. The reduced speed values are computed under the assumption of constant velocity but, since the velocity along the path is not constant, it is necessary to modify the profile in order to compensate the effects of acceleration and deceleration along the whole trajectory. This is done through the look-ahead algorithm described in the following.

The elaboration of all the points of the profile by means of numerical methods would be computationally very expensive. So, some significant points are extracted from the profile and processed, namely, the points corresponding to local minima of the velocity of the end-effector, together with other points adequately sampled from the trajectory. The need to keep the points with minimum velocity is essential in order to maintain information on the critical points of the trajectory. Furthermore, the local minima of the velocities occur when the path changes direction or the robot is close to a singular configuration. Starting from the previously extracted samples of velocity, the velocity profile is updated, so as to take into account joint accelerations limits. Starting from every point corresponding to a local minimum of the velocity, new velocity values for adjacent points are computed according to the following scheme:

1. calculate the time to travel between two consecutive points, based on the average speed $\hat{v}_{k,k+1}$ and travel time Δt_k , which are evaluated as:

$$\hat{v}_{k,k+1} = \frac{v_{t,k+1} + v_{t,k}}{2}; \quad \Delta t_k = \frac{s_{k+1} - s_k}{\hat{v}_{k,k+1}}; \quad (5)$$

where $v_{t,k}$ and s_k are the values of the end-effector velocity and arc length of the path at the k -th time instant, respectively.

2. compute the average acceleration \ddot{q}_k over the same time interval Δt_k as:

$$\ddot{q}_k = \frac{\dot{q}_{k+1} - \dot{q}_k}{\Delta t_k} \quad (6)$$

3. redefine the tangential velocity at the $k+1$ -th time step as:

$$v_{t,k+1} = \frac{v_{t,k+1} - v_{t,k}}{\gamma_{acc}} + v_{t,k}; \quad \text{with } \gamma_{acc} = \max \left| \frac{\ddot{q}_k}{\ddot{q}_{lim}} \right|; \quad (7)$$

The new velocity at the $k+1$ -th instant is reduced by a quantity proportional to the ratio γ_{acc} between the current joint acceleration at the k -th instant and the corresponding limit acceleration.

4. recalculate the end-effector velocity v_{k+1} and the joint velocity \dot{q}_{k+1} at the $k+1$ -th instant as:

$$\mathbf{v}_{k+1} = \mathbf{v}_{k+1,0} \frac{v_{t,k+1}}{v_c} \quad \dot{\mathbf{q}}_{k+1} = \mathbf{J}^{-1}(\mathbf{q}_{k+1}) \mathbf{v}_{k+1} \quad (8)$$

where \mathbf{J} is the Jacobian matrix of the robot and $\mathbf{v}_{k+1,0}$ is the end-effector velocity before the motion profile updating operation.

This algorithm is first run rightwards, beginning from the first minimum point of the speed profile, by taking into consideration only the speed ramps for acceleration (Fig. 1, left). Once the entire profile has been updated in the right direction, the algorithm is then run leftwards, so as to impose the constraints on the deceleration ramps (Fig. 1, right).

The updated velocity profile guarantees the respect of the acceleration limits at the joints but might present irregularities that could make the resulting trajectory not smooth. In order to reduce these irregularities, a moving average filter with a centered window is used. The obtained velocity profile is defined in the space domain. It can be converted to the time domain by plotting the velocity as a function of time. Knowing the distance between two consecutive points along the path, and the velocities in correspondence of these points, one can compute the time required to travel the distance between the samples:

$$t_{k+1} = t_k + \frac{s_{k+1} - s_k}{\hat{v}_{k,k+1}} \quad (9)$$

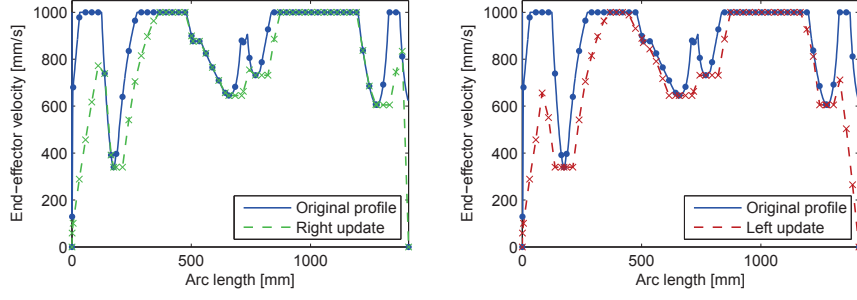


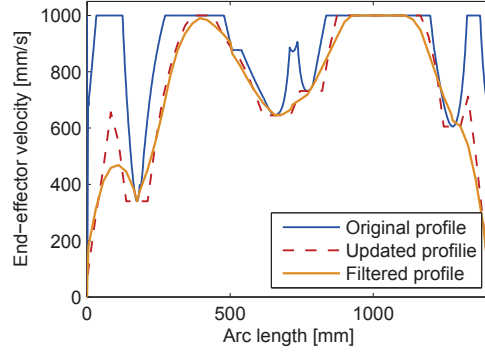
Fig. 1 Update in the right and left direction

Thus the velocity profile in the time domain can be linearly interpolated as:

$$v(t) = \frac{t - t_k}{t_{k+1} - t_k} (v_{k+1} - v_k) + v_k \quad (10)$$

At this stage, the velocity profile consists of a succession of ramps: hence, the acceleration profile is piecewise constant and the jerk results unbound. In order to limit the jerk, a moving average filter is applied to the velocity profile. This filter is similar to that used in the previous step, with the only difference that, instead of centering the buffer of samples at the output sample, in this case the buffer collects only the backward velocity samples. In short, the filter in the time domain limits the jerk and smooths the discontinuities left by the filter in the space domain, as shown in Fig. 2.

Fig. 2 Filtering of the motion profile



3 Experimental tests and results

To verify the effectiveness of the look-ahead algorithm, a model of the robot has been developed in Matlab environment. Several trajectories have been tested, in order to find the best configuration of the algorithm, by evaluating its effectiveness in limiting the joint accelerations.

The experimental tests were run on a GR680 robot, a 6 d.o.f. manipulator for painting applications, designed and manufactured by CMA Robotics. The manipulator has a non-spherical wrist. In Fig. 3 the robot used in the experimental test and its non-spherical wrist are shown. The non-spherical wrist has been adopted in order to avoid the wrist singularities, since in painting applications since almost all trajectories are programmed in the end-effector space and it is difficult to foresee when singularities may occur.

Fig. 3 The painting robot GR680



The trajectory adopted in the experimental tests is made of a single B-spline curve, interpolating 17 points. It is 9.16 m long, arranged on a vertical plane spaced 2000 mm from the base of the robot. The painting robot moves along it with a velocity limit of 1000 m/s and a joint acceleration limit equal to 540 deg/s^2 . These limits ensure that the proposed method and the manufacturer's proprietary methods lead to trajectories with similar time durations. Fig. 4 shows the capability of the method to keep the end-effector velocity below the desired peak level, a feature that cannot be ensured by the standard planning method. Moreover, in order to evaluate the performance of the proposed methodologies, the motor torques at the joints were measured. The results yielded by the look-ahead algorithm have been compared to those obtained using the original robot controller. The measure torque for the first three axes of the robot, i.e. the one that take the largest effort, are shown in fig. 5. In both cases, by adopting the look-ahead method, a noticeable reduction of the motor torques can be noticed.

Fig. 4 Velocity profile: original and with the look-ahead algorithm.

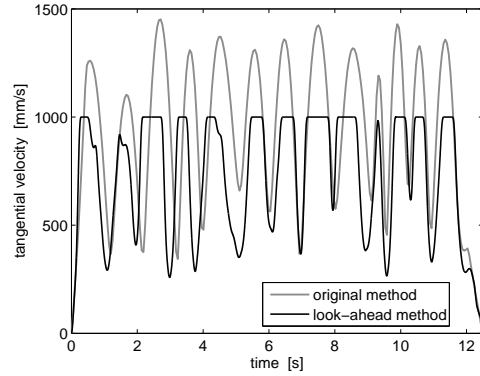
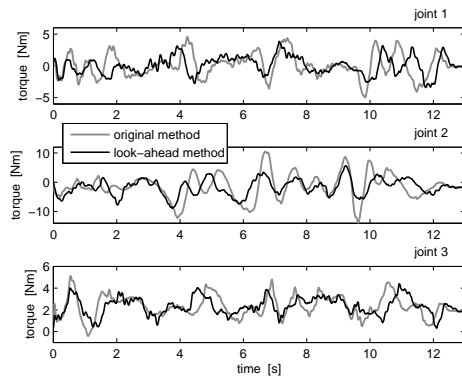


Fig. 5 Torque of second axes motor with angular acceleration limit of 540 deg/s^2 .



Experimental tests lead to the conclusion that the developed look-ahead method can limit the acceleration values at the joints, and allows to reduce the values of the motor torques at the robot joints. The reduction of the RMS value of the motor torque can reach values as high as 30% when compared to the original manufacturer's proprietary method.

4 Conclusion

In this paper, a new algorithm aimed at limiting the accelerations of joints in robots with non-spherical wrist has been presented. With respect to CNC machines, where accelerations in the end-effector space are linearly correlated with those in the joint space, robots with non-spherical wrist, characterized by a kinematics not solvable in closed form, require numerical methods in order to limit the joint accelerations, as well as to obtain a velocity profile suitable for complex tasks such as spray painting. The proposed trajectory planning is based on the parametrization of the end-effector

path by means of a B-spline curves. A model of the robot has been implemented in Matlab and used to verify the effectiveness of the method for several paths. The generated trajectories has then been tested on an industrial spray painting robot. The results of the experimental tests have shown that the algorithm allows to improve the performance with respect to previously used methods, and present a good capability in limiting joint speed and accelerations. A sensible reduction of the motor effort has been reported as well.

References

1. Lee CS (2010) Generation of velocity profiles with speed limit of each axis for high-speed machining using look-ahead buffer. *International Journal of Precision Engineering and Manufacturing*, 11(2):201–208
2. Luo FY, Zhou YF, Yin J (2007) A universal velocity profile generation approach for high-speed machining of small line segments with look-ahead. *The International Journal of Advanced Manufacturing Technology* 35(5–6):505–518
3. Hu J, Xiao L, Wang Y, Wu Z (2006) An optimal feedrate model and solution algorithm for a high-speed machine of small line blocks with look-ahead. *The International Journal of Advanced Manufacturing Technology* 28(9–10):930–935
4. Ye P, Shi C, Yang K, Lv Q (2008) Interpolation of continuous micro line segment trajectories based on look-ahead algorithm in high-speed machining. *The International Journal of Advanced Manufacturing Technology* 37(9–10):881–897
5. Shi C, Ye P (2011) The look-ahead function-based interpolation algorithm for continuous micro-line trajectories. *The International Journal of Advanced Manufacturing Technology* 54(5–8):649–668
6. Yau HT, Wang JB (2007) Fast Bezier interpolator with real-time lookahead function for high-accuracy machining. *International Journal of Machine Tools and Manufacture* 47(10):1518–1529
7. Lin MT, Tsai MS, Yau HT (2007) Development of a dynamics-based NURBS interpolator with real-time look-ahead algorithm. *International Journal of Machine Tools and Manufacture* 47(15):2246–2262
8. Wang L, Cao J (2012) A look-ahead and adaptive speed control algorithm for high-speed CNC equipment. *The International Journal of Advanced Manufacturing Technology* 63(5–8):705–717
9. Lee AC, Lin MT, Pan YR, Lin WY (2011) The feedrate scheduling of NURBS interpolator for CNC machine tools. *Computer-Aided Design* 43(6):612–628
10. Scalera L, Mazzon E, Gallina P, Gasparetto A (2017) Airbrush Robotic Painting System: Experimental Validation of a Colour Spray Model. In: *Proceedings of the International Conference on Robotics in Alpe-Adria Danube Region RAAD 2017*, Turin, Italy 21–23 June 2017
11. Chen H, Fuhlbrigge T, Li X (2008) Automated industrial robot path planning for spray painting process: a review. In: *Proceedings of the 2008 IEEE International Conference on Automation Science and Engineering (CASE 2008)*, Arlington, VA, USA 23 – 26 August 2008
12. Gasparetto A, Vidoni R, Pillan D, Saccavini E (2012) Automatic path and trajectory planning for robotic spray painting. In: *Proceedings of ROBOTIK 2012; 7th German Conference on Robotics*, Munich, Germany 21 – 22 May 2012s